OPEN ACCESS

University of BRISTOL

Xu, R., Koçak, T., Woodward, G., & Morris, KA. (2009). Bidirectional fano algorithm for high throughput sequential decoding. In *IEEE 20th Personal, Indoor and Mobile Radio Communication Conference 2009 (PIMRC2009), Tokyo, Japan* (pp. 1809 - 1813). Institute of Electrical and Electronics Engineers (IEEE). https://doi.org/10.1109/PIMRC.2009.5450140

Peer reviewed version

Link to published version (if available):
10.1109/PIMRC.2009.5450140

Link to publication record in Explore Bristol Research
PDF-document

University of Bristol - Explore Bristol Research

General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available: http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/

Bidirectional Fano Algorithm for High Throughput Sequential Decoding

Ran Xu*, Taskin Kocak*, Graeme Woodward†, Kevin Morris* and Craig Dolwin†

*Centre for Communications Research
Merchant Venturers Building
University of Bristol
Bristol, UK
{Ran.Xu, T.Kocak, Kevin.Morris}@bristol.ac.uk

†Toshiba Research Europe Limited
Telecommunications Research Laboratory
Queens Square
Bristol, UK
{Graeme.Woodward, Craig.Dolwin}@toshiba-trel.com

Abstract—**Various techniques, such as bidirectional search, have been employed in sequential decoding to reduce the decoding delay. In this paper, a bidirectional Fano algorithm (BFA) is proposed, in which a forward decoder (FD) and a backward decoder (BD) search in the opposite direction simultaneously. It is shown that the proposed BFA can reduce the average decoding delay by at least 50% compared to the unidirectional Fano algorithm (UFA). Due to the reduction in the variability of the computational effort by using bidirectional search, there is even higher decoding throughput improvement at low signal-to-noise-ratio (SNR). For example at Eb/No=3dB, there is 300% throughput improvement by using the BFA decoding compared to the conventional UFA decoding. The proposed BFA decoding technique can be employed in very high throughput wireless communication systems with low hardware complexity and power consumption.**

Keywords-bidirectional decoding; Fano algorithm; high throughput decoding; sequential decoding

I. INTRODUCTION

The Viterbi algorithm (VA) is known to achieve maximum likelihood (ML) decoding for convolutional codes [1]. However, the decoding complexity of the VA increases exponentially with respect to the constraint length. For this reason it is desirable to find lower complexity decoding approaches for more complex codes, particularly with constraint lengths greater than eight.

The concept of sequential decoding was first introduced by Wozencraft in 1957 [2]. Later in 1963, Fano proposed the sequential decoding algorithm which has become known as the Fano algorithm [3]. Thereafter, Zigangirov and Jelinek developed the Stack algorithm independently in [4] and [5], respectively. This is also known as the ZJ algorithm. Unlike the VA, sequential decoding only explores partial paths locally in the code tree, so the decoding complexity can be reduced at medium to high SNR. Since its decoding complexity is independent from the constraint length, sequential decoding can achieve a better bit-error-rate (BER) performance for a given decoder complexity by employing a longer constraint length for the convolutional code. More recent discussions on sequential decoding algorithms can be found from [6]-[9].

Because it has low storage and sorting requirements, the Fano algorithm is considered to be more practical for hardware implementations compared to the Stack algorithm [10]. A Fano decoder was implemented with quasi-delay-insensitive (QDI) templates, which runs at 430MHz and consumes only 32mW by using the TSMC 0.25-µm CMOS technology [11]. It was shown in [12] that a reconfigurable Fano decoder which has a maximum decoding throughput of 75Mb/s takes only 12.28% of the equivalent resources of the Viterbi decoder in [13], both of which were implemented on the same Virtex-4 FPGA device.

The computational effort of sequential decoding is random and a buffer is always required to store the codewords to be decoded. A buffer overflow may be caused by a long decoding delay, and the corresponding frame will be erased. Various techniques were introduced in sequential decoding to reduce the decoding delay, thus to reduce the buffer size and the erasure probability and increase the decoding throughput. For example in the Fano algorithm, the threshold increment value Δ can be increased or a trace-back limitation can be imposed to reduce the decoding delay, which will sacrifice some BER performance [1].

Due to the availability of several GHz of unlicensed bandwidth around 60GHz, multi-gigabit per second wireless communication has become a popular research topic [14]. WirelessHD is one of the standards targeting the 60GHz radio applications [15]. To achieve the high throughput required, the WirelessHD draft standard proposes simultaneous transmission of eight interleaved codewords, each encoded by a convolutional code. It is straightforward to use eight parallel independent Viterbi decoders. However, the hardware cost will be eight times of a single Viterbi decoder. Therefore, it is desirable to reduce the hardware cost and also the power consumption, which leads to the consideration of Fano decoding. Compared with the VA, sequential decoding has the advantage of very low hardware complexity and power consumption, and it can decode very long constraint length convolutional code. But it is difficult for sequential decoding to achieve high decoding throughput due to its irregular decoding operations and its variability in decoding delay. In this work, a bidirectional Fano decoding algorithm (BFA) is proposed which can reduce the decoding delay and improve the decoding

This work is sponsored by Telecommunications Research Laboratory of Toshiba Research Europe Limited.

Authorized licensed use limited to: UNIVERSITY OF BRISTOL. Downloaded on May 21,2010 at 10:11:23 UTC from IEEE Xplore. Restrictions apply.

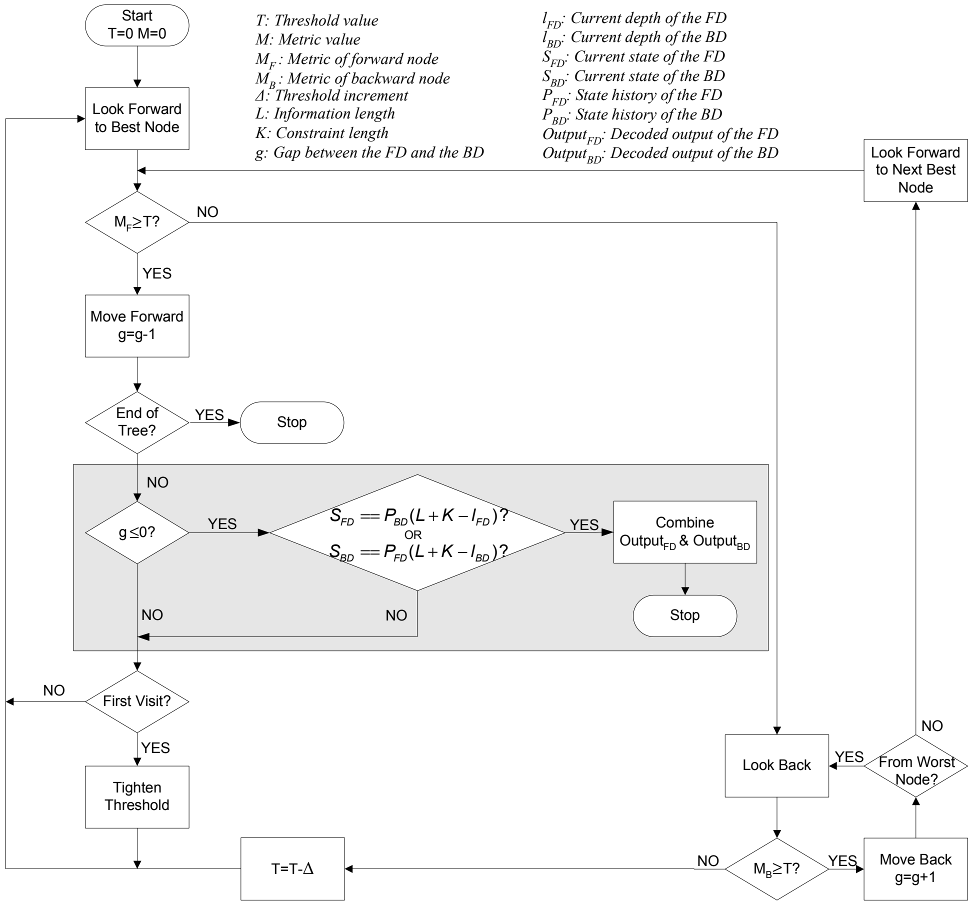

Figure 1. Flow chart of the FD or the BD with a bidirectional Fano algorithm

throughput compared to the conventional unidirectional Fano algorithm (UFA). The rest of the paper is organized as follows. Firstly, the principles of the BFA are introduced in Section II. How to make a trade-off between error rate performance and decoding throughput will be discussed. Simulation results and analysis are given in Section III. The BFA is compared with the UFA in terms of BER, computational variability and throughput. The conclusions are drawn in Section IV.

II. Bidirectional Fano Algorithm

A. *Bidirectional Sequential Decoding Algorithms*

The idea of bidirectional search in sequential decoding was first proposed by Forney in 1967 [16]. A forward decoder and a backward decoder search a terminated tree from the start state and the end state simultaneously. The search terminates when either of the two decoders reaches the end of the code tree. However, it was shown that this scheme could not improve the computational performance much. K. Li and S. Kallel [17] introduced an efficient bidirectional sequential decoding technique based on the Stack algorithm. It was shown that the proposed algorithm could double the Pareto exponent[1] and the reduced computational variability can decrease the erasure probability substantially. A bidirectional multiple stack algorithm was developed in [19] by the same authors. Independently, V. Senk and P. Radivojac also proposed a bidirectional stack algorithm in [20]. An improved version was demonstrated in [21], which was not only faster than the bidirectional sequential decoding in [17] but also than that introduced in [20].

However, as discussed above in Section I, the sorting operation and the large memory requirement in the Stack algorithm make it difficult to achieve high decoding throughput. The Fano algorithm is more suitable for hardware implementations and can achieve high decoding throughput. It is expected that if the bidirectional searching technique could be applied to the Fano decoding, the computational variability and the average decoding delay will be reduced and the decoding throughput will be improved.

B. *Bidirectional Fano Algorithm*

Similar to the bidirectional sequential decoding proposed in [17], there is a forward decoder (FD) and a backward decoder (BD) in the bidirectional Fano algorithm. Both of them start decoding from the known state zero and perform decoding in the forward and backward direction in parallel as shown in Fig. 2. The decoding will finish if the FD and the BD merge somewhere in the code tree. Otherwise, if the FD and the BD

[1] The Pareto exponent is a parameter of a Pareto distribution, which in this case describes the probability of a computational effort exceeding a given value. A higher exponent means that the variability in the computational effort is smaller [18].

Authorized licensed use limited to: UNIVERSITY OF BRISTOL. Downloaded on May 21,2010 at 10:11:23 UTC from IEEE Xplore. Restrictions apply.

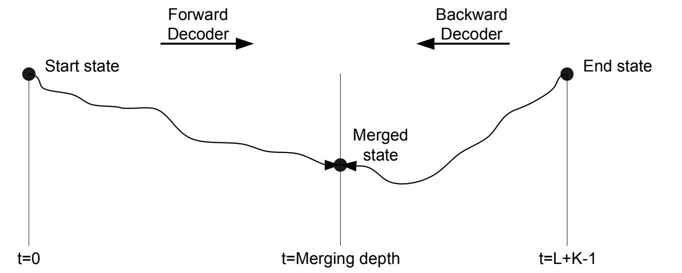

Figure 2. Illustration of bidirectional Fano decoding, where L is the information length and K is the constraint length of the convolutional code

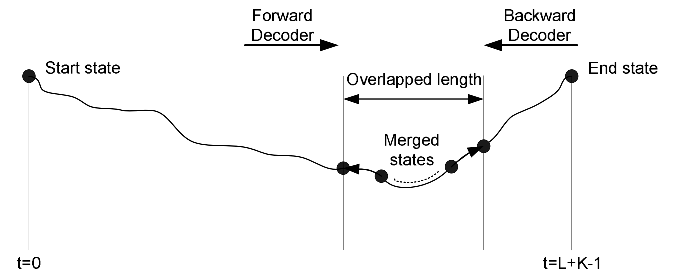

Figure 3. Illustration of the more rigorous merging check in the proposed BFA, in which there are more than one merged states and the overlapped length should be not less than 1

cannot merge, the decoding will finish when either of them reaches the other end of the code tree in its own direction. The concept of merging is defined in the same manner as [17], which means that the FD and the BD have the same encoder state and the same position within the codeword. The flow chart of the FD within the BFA is shown in Fig. 1. The flow chart of the BD is analogous with that of the FD, and the variables l_{FD}, l_{BD}, S_{FD}, S_{BD}, P_{FD}, P_{BD} and $Output_{FD}$, $Output_{BD}$ need to be appropriately transposed.

Both of the FD and the BD decode the same codeword in the opposite direction according to the flow chart in Fig. 1, which is similar to the conventional unidirectional Fano algorithm (UFA) [1], except that a merging check is carried out after a forward movement is made. A variable g is defined as the gap between the depth of the FD and that of the BD. The initial value of g is set as $L+K-1$. It is decreased by one ($g=g-1$) because of the forward movement of either the FD or the BD, and it is increased by one ($g=g+1$) because of the backward movement of either the FD or the BD. The check merging operation is enclosed in the shaded box in Fig. 1 which may be evoked by the forward movement of either the FD ($l_{FD}=l_{FD}+1$) or the BD ($l_{BD}=l_{BD}+1$). If the current state (S_{FD} or S_{BD}) is the same as the state in the other decoder's state history at the same depth ($S_{FD}=P_{BD}(L+K-l_{FD})$ or $S_{BD}=P_{FD}(L+K-l_{BD})$), a merging is detected and the decoded output of the FD and that of the BD should be combined as the final output. In order to check the merging condition, both of the FD and the BD need to maintain a path history (P_{FD} and P_{BD}), which stores all the past states in the path leading to the current state. The size of the output buffer also needs to be doubled, because the FD and the BD may not have the same decoded output. But the path history and the additional buffer require only a small amount of extra memory.

To increase the reliability of the merge operation, and thus improve the error rate performance slightly, a modification to the simple merge strategy described above is to require that two or more consecutive states are identical at the same depths within the codeword for the FD and the BD, which is shown in Fig. 3. The improved BER performance is at the expenses of higher computational effort which will be shown in the next section.

III. SIMULATION RESULTS

The performance of the proposed BFA is examined by simulations in this section. The convolutional code in the simulations is the same as the one used in the WirelessHD specification [15]. The code rate is R=1/3. The generator polynomials are $g_0=\{133\}_8$, $g_1=\{171\}_8$ and $g_2=\{165\}_8$, and the constraint length is K=7. The threshold increment value in the Fano algorithm is $\Delta=2$, the metric calculation was based on the Fano metric which is considered to be optimal for sequential decoding [1], and the output of the demodulator was 1-bit hard decision. The modulation was BPSK and the channel used was AWGN channel.

Firstly, the merging depth distribution of the BFA was investigated. By transmitting N=50,000 frames with each frame having 200 information bits plus 6 tail bits which return the encoder back to state zero, there were 55, 38 and 8 frames which did not merge in the BFA decoding at Eb/No=3dB, 4dB and 5dB, respectively. However, all of the frames merged in the BFA decoding at 6dB. It is shown in Fig. 4 that the merging depths of all the merged frames have an approximate Gaussian distribution, and the variance decreases as the SNR increases. The BER performance of the BFA is also compared with that of the UFA. It is shown in Fig. 5 that at BER=10^{-4} the BFA has a penalty of about 0.1dB compared to the UFA and about 0.2dB compared to the VA.

The advantage of the BFA can be seen in Fig. 6 and Fig. 7. The term *Number of Iterations* (NoI) was adopted to measure the computational effort. With reference to the flowchart in Fig. 1, an iteration is defined as the execution from the Look Forward to Best Node (LFB)/Look Forward to Next Best Node (LFNB) to the next LFB/LFNB. The x-axis (Total NoI) in Fig. 6 is the sum of the NoI of the FD and that of the BD for the BFA:

$$NoI_{total} = NoI_{FD} + NoI_{BD} \tag{1}$$

It is shown in Fig. 6 that the computational variability of the BFA is less that of the UFA. The computational effort of the VA in terms of NoI has a fixed value:

$$NoI_{VA} = (L+K-1)\times 2^{K-1} \tag{2}$$

In Fig. 7 the average decoding throughput was compared between the BFA and the UFA for different numbers of merged states at different SNR values. In the BFA, the decoding delay of each codeword is determined by the maximum NoI of the FD and that of the BD:

$$Delay_{BFA} = \max(NoI_{FD}, NoI_{BD}) \tag{3}$$

Authorized licensed use limited to: UNIVERSITY OF BRISTOL. Downloaded on May 21,2010 at 10:11:23 UTC from IEEE Xplore. Restrictions apply.

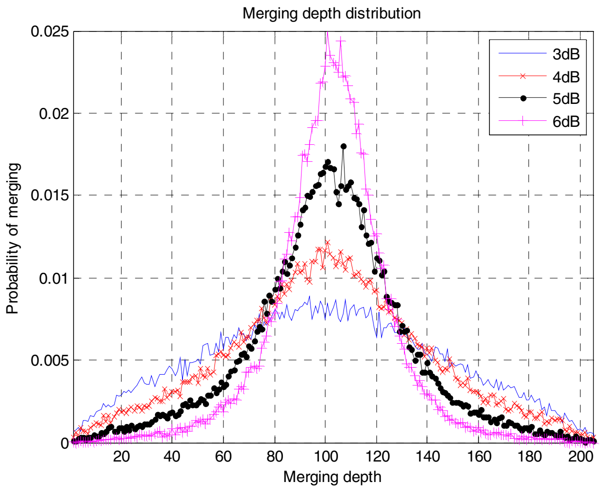

Figure 4. Merging depth distribution of the BFA

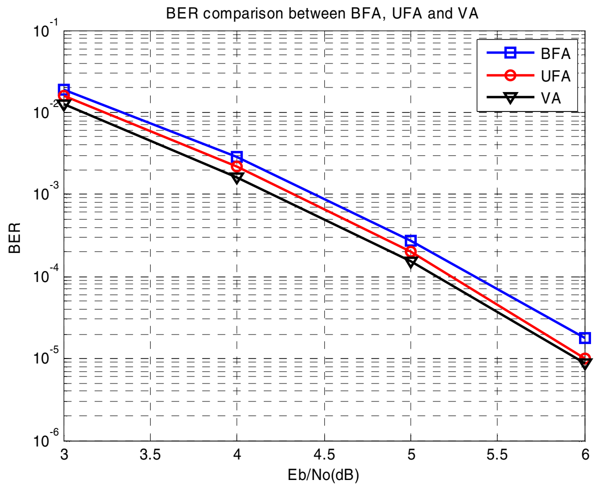

Figure 5. BER performance comparison between the VA, the UFA and the BFA at Eb/No=3 to 6dB in the AWGN channel

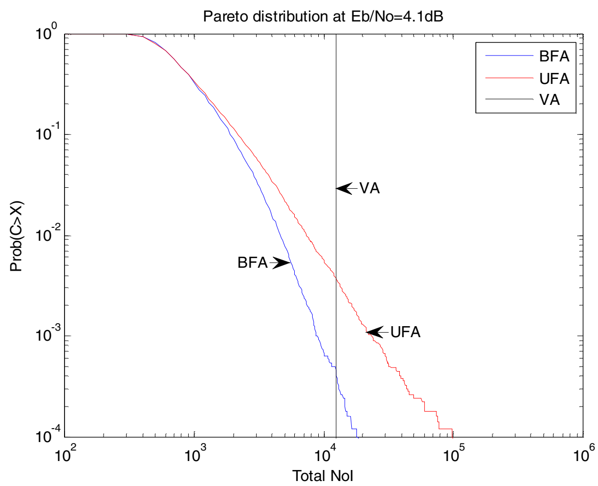

Figure 6. Pareto distribution of the BFA, the UFA and the VA at Eb/No=4.1dB corresponding to the Pareto exponent of 1. The information length is 200 bits.

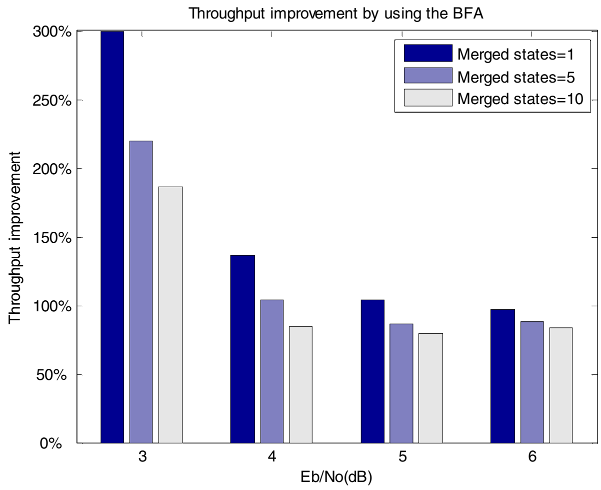

Figure 7. Throughput improvement by using the BFA with respect to the UFA for different number of merged states

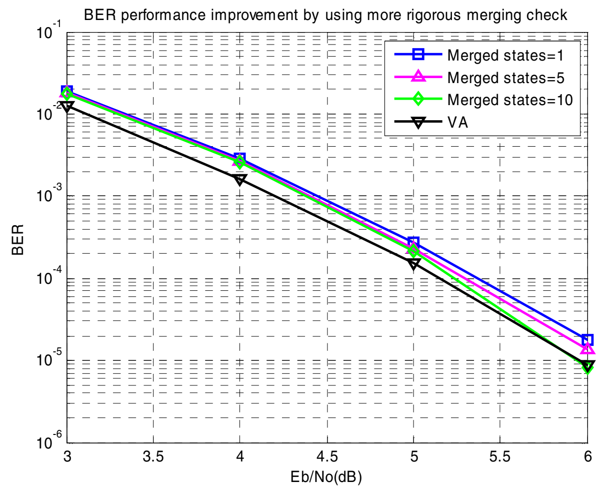

Figure 8. BER performance improvement by using more rigorous merging check in the BFA with the number of merged states of 1, 5, and 10 at Eb/No=3 to 6dB in the AWGN channel

The average NoI is proportional to the average decoding delay. Equivalently, the average decoding throughput is inversely proportional to the average NoI. The metric 'throughput improvement' is adopted to compare the decoding throughput of the UFA and that of the BFA, and it is measured by simulation of N codewords:

$$\text{Throughput Improvement} = \left(\frac{\sum_{i=1}^{N} Delay_{i,UFA} / N}{\sum_{i=1}^{N} Delay_{i,BFA} / N} - 1\right) \times 100\% \quad (4)$$

where $Delay_{i,UFA}$ and $Delay_{i,BFA}$ are the delays in terms of NoI to decode each codeword by using the UFA and the BFA, respectively. Simulation results are presented for N=50,000 experiments. It can be seen from Fig. 7 that for the information length of 200 bits and Eb/No=3dB, the throughput improvement is 300% by using the BFA with one merged state. But the improvement decreases as the SNR increases. For example at Eb/No=6dB, the throughput improvement drops to about 100% and it converges to this value at higher SNR

Authorized licensed use limited to: UNIVERSITY OF BRISTOL. Downloaded on May 21,2010 at 10:11:23 UTC from IEEE Xplore. Restrictions apply.

values. This is because the FD and the BD tend to merge in the middle of the code tree at high SNR as shown in Fig. 4.

It is shown in Fig. 8 that as the number of merged states increases, the BER performance of the BFA can approach that of the UFA and the VA. For example, at Eb/No=5.2dB, the BER of the BFA with 10 merged states is the same as that of the UFA in Fig. 5, and there is only 0.1dB coding gain loss compared to the VA. However, the improvement of the BER performance is at the cost of loss in throughput improvement as shown in Fig. 7. The throughput improvement decreases as the number of merged states increases at each SNR. But a BFA with a more rigorous merging check condition can still achieve higher decoding throughput than the UFA. For example, at Eb/No=6dB, there is still 83% throughput improvement for the BFA with 10 merged states than the UFA.

IV. CONCLUSION

In this paper, a bidirectional Fano decoding algorithm for convolutional codes was proposed. It has been shown that the BFA can reduce the decoding delay by applying a forward decoder and a backward decoder to search in the opposite direction simultaneously. The average decoding delay can be reduced by at least 50% compared to unidirectional Fano decoding and there is higher throughput improvement at low SNR. The proposed bidirectional Fano decoding technique is very attractive for high throughput wireless communication systems, such as the WirelessHD system, due to its low hardware complexity, low power consumption and ability to decode very long constraint length convolutional codes.

ACKNOWLEDGMENT

The authors would like to thank the directors at Telecommunications Research Laboratory of Toshiba Research Europe Ltd. for supporting this work.

REFERENCES

[1] S. Lin and D. J. Costello, Jr., *Error Control Coding: Fundamentals and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1983.

[2] J. M. Wozencraft, "Sequential decoding for reliable communications," *IRE Nat. Conv. Rec.*, vol. 5, pt. 2, pp. 11-25, 1957.

[3] R. M. Fano, "A heuristic discussion of probabilistic decoding," *IEEE Transactions on Information Theory*, vol. IT-9, no. 2, pp. 64-74, Apr. 1963.

[4] K. Sh. Zigangirov, "Some sequential decoding procedures," *Probl. Pered. Inform.*, vol. 2, no. 4, pp. 13-15, 1966.

[5] F. Jelinek, "Fast sequential decoding using a stack," *IBM J. Res. Devel.*, vol. 13, pp. 675-685, Nov. 1969.

[6] A. D. Murugan, H. El Gamal, M. O. Damen and G. Caire, "A unified framework for tree search decoding: Rediscovering the sequential decoder," *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 933-953, Mar. 2006.

[7] J. Hagenauer and C. Kuhn, "The list-sequential (LISS) algorithm and its application," *IEEE Transactions on Communications*, vol. 55, no. 5, pp. 918-928, May 2007.

[8] B. Narayanaswamy, R. Negi and P. Khosla, "An analysis of the computational complexity of sequential decoding of specific tree codes over Gaussian channels," *IEEE International Symposium on Information Theory - Proceedings*, pp. 2508-2512, 2008.

[9] M. Sikora and D. J. Costello, "Supercode heuristics for tree search decoding," *IEEE Information Theory Workshop*, pp. 411-415, 2008.

[10] J. B. Anderson and S. Mohan, "Sequential coding algorithms: A survey and cost analysis," *IEEE Transactions on Communications*, vol. COM-32, pp. 169-176, Feb. 1984.

[11] R. O. Ozdag and P. A. Beerel, "An asynchronous low-power high-performance sequential decoder implemented with QDI templates," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 14, no. 9, pp. 975-985, Sep. 2006.

[12] M. Benaissa and Y. Zhu, "Reconfigurable hardware architectures for sequential and hybrid decoding," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, no. 3, pp. 555-565, Mar. 2007.

[13] M. Benaissa and Y. Zhu, "A novel high speed configurable Viterbi decoder for broadband access," *EURASIP J. Appl. Signal Process.*, vol. 2003, no. 12, pp. 1317-1327, Dec. 2003.

[14] H. Singh, J. Oh, C. Kweon, X. Qin, H.-R. Shao and C.Ngo. "A 60 GHz wireless network for enabling uncompressed video communication," *IEEE Communications Magazine*, vol. 46, no. 12, pp. 71-78, 2008.

[15] "Wireless High-Definition (WirelessHD)"; http://www. wirelesshd.org/

[16] G. D. Forney, Jr., "Final report on a coding system design for advanced solar missions," Rep. NAS2-3637, Contract NASA CR73167, NASA Ames Res. Ctr., Moffett Field, CA, 1967.

[17] S. Kallel and K. Li, "Bidirectional sequential decoding," *IEEE Transactions on Information Theory*, vol. 43, no. 4, pp. 1319-1326, Jul. 1997.

[18] F. Jelinek, "Upper bounds on sequential decoding performance parameters," *Transactions on Information Theory*, vol. 20, no. 2, pp. 227-239, Mar. 1974.

[19] K. Li and S. Kallel, "Bidirectional multiple stack algorithm," *IEEE Transactions on Communications*, vol. 47, no. 1, pp. 6-9, Jan. 1999.

[20] V. Senk and P. Radivojac, "Bidirectional stack algorithm," *IEEE International Symposium on Information Theory - Proceedings*, pp. 500, 1997.

[21] V. Senk and P. Radivojac, "A new bidirectional algorithm for decoding trellis codes," *EUROCON'2001, Trends in Communications, International Conference on*, vol. 1, pp. 34-36, Jul. 2001.

Authorized licensed use limited to: UNIVERSITY OF BRISTOL. Downloaded on May 21,2010 at 10:11:23 UTC from IEEE Xplore. Restrictions apply.